*In the Specification*

Please replace the corresponding paragraphs with the following:

[0038] FIG. 1A shows a basic system configuration in which the present invention may be practiced in accordance with one embodiment thereof. Documents or files, such as product descriptions, customer lists and price schedules, may be created using an authoring tool executed on a client computer 100, that may be a desktop computing device, a laptop computer, or a mobile computing device. Exemplary authoring tools may include Microsoft Office® (e.g., Microsoft Word®, Microsoft PowerPoint®, and Microsoft Excel®), Adobe FrameMaker® and Adobe Photoshop®.

[0042] In another setting, the computer 100 and the computing or storage device 102 are inseparable, in which case the computing or storage device 102 may be a local store to retain secured documents or receive secured network resources (e.g., dynamic Web contents, results of a database query, or a live multimedia feed). Regardless of where the secured documents or secured sources are actually located, a user, with a proper access privilege, can access the secured documents or sources from the computer 100 or the device 102 using an application (e.g., Internet Explorer®, Microsoft Word® or Acrobat Reader®).

[0051] Referring now to FIG. 2A, an illustration diagram of securing a created document 200 is shown. After the document 200 is created with an application or authoring tool (e.g., Microsoft ~~WORD~~ Word®), upon an activation of a "Save," "Save As" or "Close" command or automatic saving invoked by an operating system, the application, itself, or an application that is previously registered with the server, the created document 200 is caused to undergo a securing process 201. The securing process 201 starts with an encryption process 202, namely the document 200 that has been created or is being written into a store is encrypted by a cipher with a file key.

In other words, the encrypted document ~~could~~ can not be opened without the file key (i.e., a cipher key).

[0057] Alternatively, a secured document in a folder appears substantially similar to a regular document and launches the same application when activated except the application ~~would~~ will fail to access the contents therein. For example, icons or file names of secured documents may appear in a different color or with a visual ~~indication~~ indicator to distinguish from non-secured documents. When a secured document ~~is~~ unintentionally ends up in a machine or readable medium (e.g., CD or disk), if a user of the machine or a machine attempting to read the readable medium has no proper user key or if the user cannot be authenticated, the secured document ~~would~~ will not be successfully accessed.

[0068] FIG. 2C.1 illustrates an exemplary structure of a secured document 236 including a header 238 and an encrypted portion 239. The header 238 permits four different [[240-243]] entities 240-243 to access the secured document 236. The four different entities 240-243 include two individual users and two group users, wherein the group users mean that everyone in [[a]] the group ~~could~~ can access the document with the same privileges. The two individual users have two different access privileges. User A can only read the document while user D can edit and read the document. While everyone in Group B can read and edit the document, everyone in Group C can only print the document. Each entity has a corresponding ID ~~to be~~ associated with the corresponding users and its own access rules. According to one embodiment, the header 238 in the secured document 236 is partitioned into corresponding four sub-headers 240-243, each designated to one user or group and keeping a file key therein and encrypted with a separate user key. In other words, when User A is requesting the secured document 236, only the header 240 designated to User A is decrypted with a user key (e.g., key A) belonging to the user

A and authenticated with the user, the rest of the sub-headers 241-243 remain encrypted. In any case, once one of the sub-headers 241-243 is decrypted, the secured document can be decrypted with a key (e.g., file key) retrieved from the decrypted sub-header.

[0077] Unlike prior art systems in which documents to be secured are encrypted by an encryption process initiated by a user, one of the features in the present invention is to activate a cipher process (i.e., encryption/decryption process) transparently as far as the user is concerned. In other words, the user is not made aware that a document is being made secured through the cipher process while being ~~wrote~~ written into a store.

[0078] FIG. 3 shows an exemplary implementation 300 of how a document securing module (DSM) 302 interacting with and operating within an operating system 304 (e.g., ~~WINDOWS~~ Windows 2000®) to ensure that a document is made secure in a manner that is transparent to the user.

[0080] In operation, a user selects a secured document that is associated with an application 306 (e.g., MS ~~WORD~~ Word®, PowerPoint®, or printing). The application 306 ~~acts~~ acting on the secure document calls an API (e.g., createFile, a Common Dialog File Open Dialog with Win 32 API in MS Windows®) to access the installable file system (IFS) ~~manger~~ manager 312. If it is detected that an "Open" request is made from the application 306, the request is passed to an appropriate file system driver (FSD) 314 to access the requested secured document. At the same time, the cipher module 310 is activated and an authenticated user key is retrieved from a local store to decrypt the header in the requested secure document. If the encrypted header is decrypted and the access rules therein are measured successfully against the user's access privileges, then a file key is retrieved from the header of the secured

document and the cipher module 310 proceeds to decrypt the encrypted document in the DSM 302. The clear contents are then returned to the application 306 through the IFS manager 312. For example, if the application 306 is an authoring tool, the clear contents are displayed. If the application 306 is a printing tool, the clear contents are sent to a designated printer.